Module 3F5: Computer and Network Systems

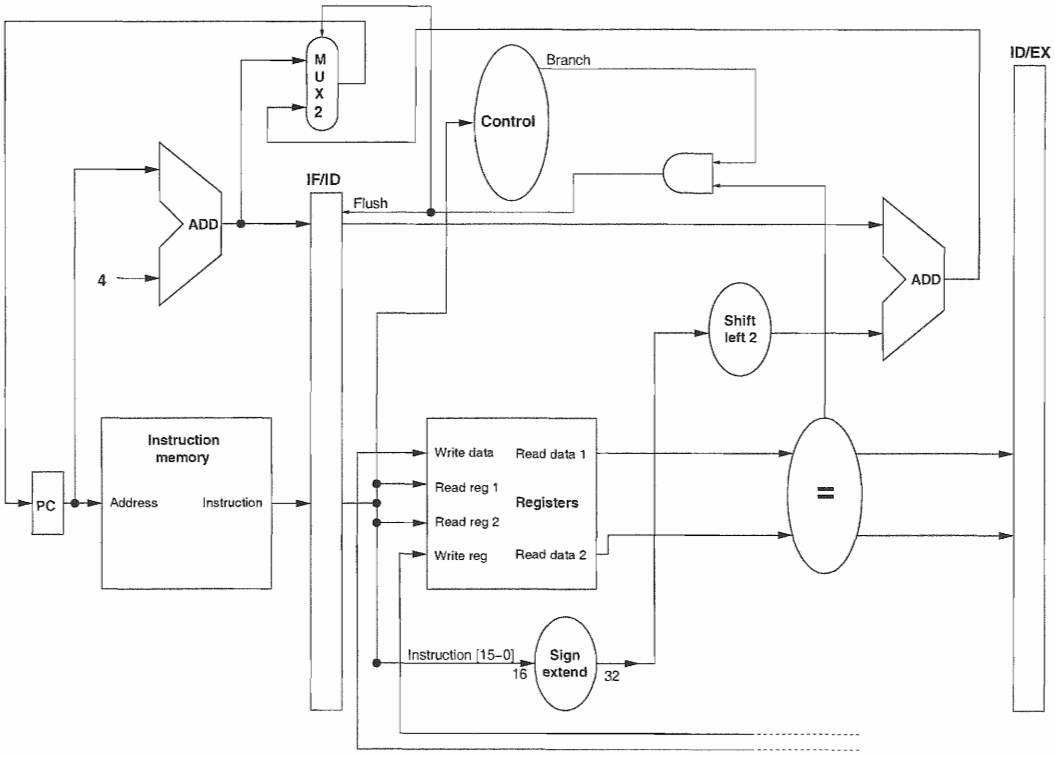
Solutions to 2009 Tripos Paper

Authors: Andrew Gee and Tim Wilkinson

1. **Pipelined datapaths and branch hazards**

(a) Branch hazards occur when the address of the next instruction is required (for instruction fetching) before an earlier conditional branch instruction has been evaluated. They can be resolved by any one of, or a combination of, (i) stalling the pipeline until the address is known, (ii) assuming the branch is not taken and then flushing the pipeline if it is, (iii) more sophisticated forms of dynamic branch prediction, followed by flushing if necessary, (iv) *delayed branches*, whereby the instruction following a branch is always executed irrespective of whether the branch is taken. [20%]

(b) In the modified datapath below, the branch target calculation is moved from the EX stage to the ID stage. The registers are also compared at the ID stage, so the branch decision is known at this point. If the branch is taken the next instruction is flushed by writing zeros into the IF/ID pipe register.



There may be some impact on the clock cycle time, since there are now cascaded functional units connected in series within the ID stage. Depending on the execution time of these units, there is a danger that ID might become the rate-limiting stage of the pipeline and the clock speed might need to be reduced. However, note that a fully functional (and hence relatively slow) ALU is not needed to compare the registers. We just need to know whether they are equal or not. This could be done by a parallel, bit-wise XOR, followed by an AND gate, with no need for any carries. Using this sort of fast comparator, in place of a full ALU, saves hardware and reduces the risk of the clock rate having to be reduced. [40%]

(c) In the tables below, t indicates that the branch is taken, while n indicates that the branch is not taken. For the first scheme:

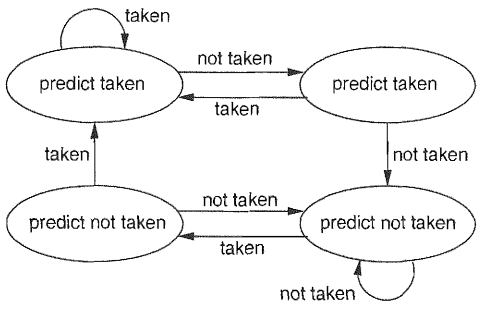
actual	t	t	t	t	t	t	t	t	t	n	t	t	t	t	t
prediction	n	t	t	t	t	t	t	t	t	t	n	t	t	t	t
correct?	✗	✔	✔	✔	✔	✔	✔	✔	✔	✗	✗	✔	✔	✔	✔

In the long run, this scheme is going to get two out of ten predictions wrong: its asymptotic accuracy is therefore 80%. For the second scheme:

actual	t	t	t	t	t	t	t	t	t	n	t	t	t	t	t
prediction	n	n	t	t	t	t	t	t	t	t	t	t	t	t	t
correct?	✗	✗	✔	✔	✔	✔	✔	✔	✔	✗	✔	✔	✔	✔	✔

In the long run, this scheme is going to get one out of ten predictions wrong: its asymptotic accuracy is therefore 90%.

The branch history needs to be stored in a table with extremely rapid access. We could use a memory indexed by the address of the branch instruction, but such a memory would need to be as large as the instruction address space. More practically, we could index by the lower portion of the address: this dramatically reduces the size of the table at the expense of a (small) risk that the indexed prediction corresponds to a different branch instruction to the one currently being executed. Scheme one requires just one bit of storage per index (t or n), while scheme two requires two bits (the states of the 4-state finite state machine below).



Any scheme with a "predict taken" option will need to know the branch target at the IF stage. So the sign extend, shift left and ADD units at the ID stage in the diagram in (b) must be moved to the IF stage. [40%]

2. **Input/Output and caches**

(a) The latency of an I/O system refers to the time required to initialize an I/O process. The bandwidth refers to the quantity of data that can flow through the system per second. The following factors contribute to the latency of hard disk I/O: seek time, rotational latency, controller overhead. [10%]

(b) Keyboard: around 10 B/s. Hard disk: around 50 MB/s. Ethernet adapter: 1 Gb/s. Graphics card: around 500 MB/s. [10%]

(c) The fact that the application is I/O limited, while most of the system's memory remains unused, suggests that the operating system does not offer a file system cache. It should exploit temporal locality of reference to the disk by caching recently accessed files in memory. It could also exploit spatial locality of reference by transferring large blocks at a time from the file to the file system cache. When all the physical memory is in use, the interaction between the file system cache and conventional virtual memory (VM) becomes an issue. Conventional VM pages should get first priority for physical memory allocation, with the file system cache shrinking as necessary. When the file system cache is full, a least recently used (LRU) replacement strategy would seem appropriate. Coherency between the file system cache and the disk file is another issue: write back would appear to offer the best performance, but then data loss becomes more likely in the event of a power interruption. [30%]

(d) PCI-e and SATA are serial point-to-point networks, while PCI and PATA are buses. The move to serial point-to-point networks arose because buses could keep up with the bandwidth of today's I/O devices. The clock rate of a parallel bus is limited by noise, stray capacitance, crosstalk and clock skew. Generally, if you want a very fast parallel bus, you'll need to make it physically short and limit the number of devices allowed to tap into it. This conflicts with the basic requirement than an I/O bus should be long and support as many I/O devices as necessary.

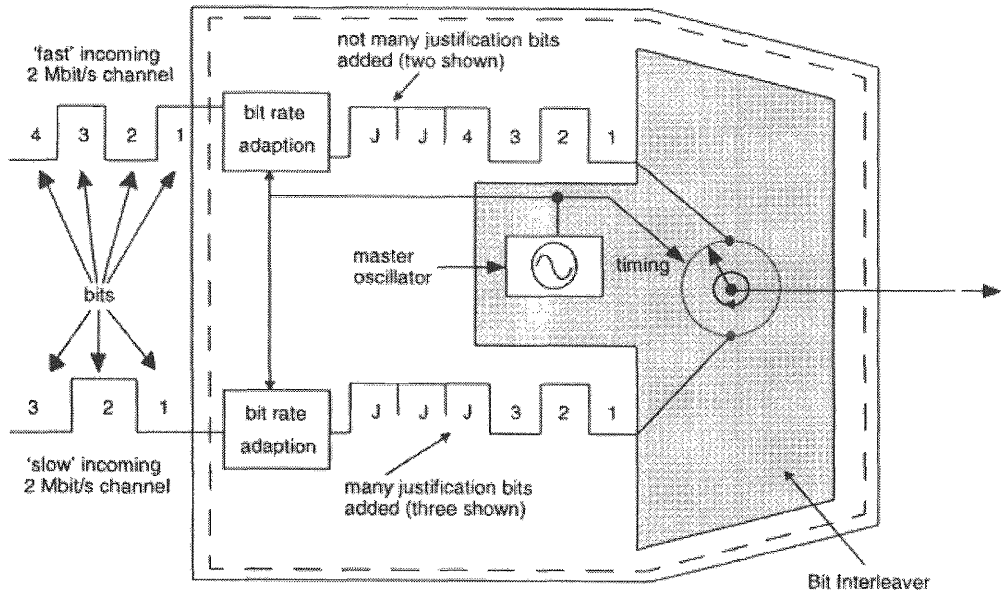
These problems are avoided by doing away with the shared, parallel bus, and replacing it with a switched point-to-point network. To keep the number of wires manageable, these networks are serial. But they can run very fast, since there are only two devices on each link — so less load, device noise and stray capacitance — and hardly any crosstalk. These new I/O networks also circumvent the need for bus mastering protocols. Data transfer is typically synchronous to an embedded clock. Serial connections have the added advantage of requiring fewer wires. This means less clutter and hence better air flow and cooling inside computer cases. [30%]

(e) The first two characteristics ensure that the encoded signal is DC-balanced with little low frequency energy. It can thus be transmitted over AC-coupled connections: these are far easier to realise than DC-coupled connections. The final characteristic ensures that binary transitions are sufficiently frequent for a phase locked loop to lock onto the signal, allowing recovering of the embedded clock. [20%]

Question 3

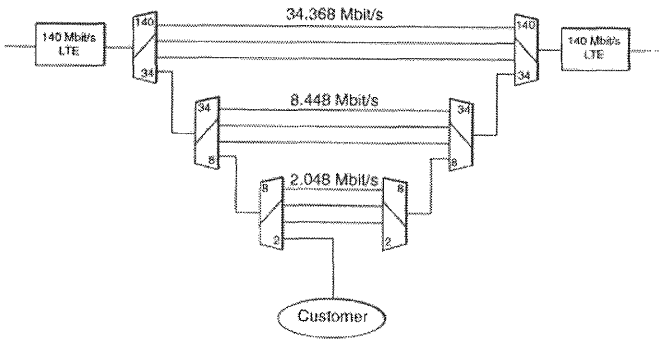
Crib (more verbose than expected)

a) Each of the digitised voice channels will be generated by different equipment each with its own clock. Hence they will have slightly different data rates. Before they are interleaved, they must undergo bit rate adaptation. All of the channels are brought to the same data rate by adding dummy information known as 'justification bits'. This is known as 'bit stuffing'.



The justification bits must be identified and discarded when the channel is DeMuxed. Hence the extra channels in PDH.

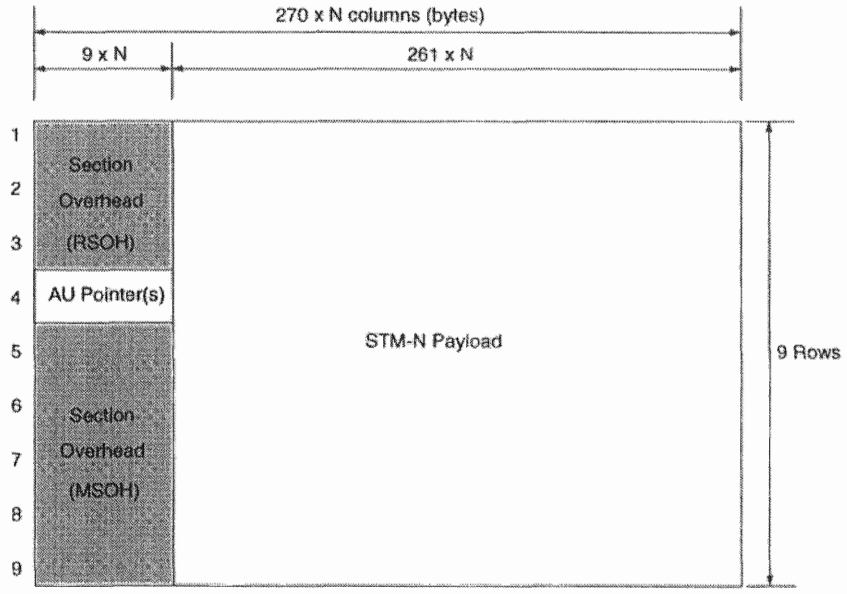
The problem occurs when a single 2Mbit channel needs to be dropped or added from a higher order Mux. (Drop and Insert). All of the channels must be demuxed and then muxed again as the position of a channel is unknown in the higher orders. ie 140 to 34, 34 to 8 and then 8 to 2Mbit/sec. This is very expensive as it requires a lot of equipment. This is the mux mountain which limits the datarate and number of channels due to the cost and complexity of the equipment.



There is also inadequate provision of network management, which is important when combining POTS and computer data.

b) The aim of SDH was to correct the defects of PDH using the following features.

- Synchronisation. All elements of the system are synchronised to the same master clock. All tributaries are at a common rate before Mux.
- Pointers. Information about the muxed signals is transmitted at fixed intervals, which indicate the positions of units within the mux process. Hence any unit within the mux process can be identified and drop and insert processes can be done dynamically.
- Control and management. Time slots are put aside for a variety of tasks in ensuring the synchronicity of voice and data services.



Positions in the frame are given to data bytes from different sources and pointers are set at the start of each frame to allow the data to be found at demux. The STM-1 frame must be capable of muxing a variety of different sources, including all of the old PDH rates. Hence there are standards to map 1.5 – 140Mbits/sec onto STM-1.

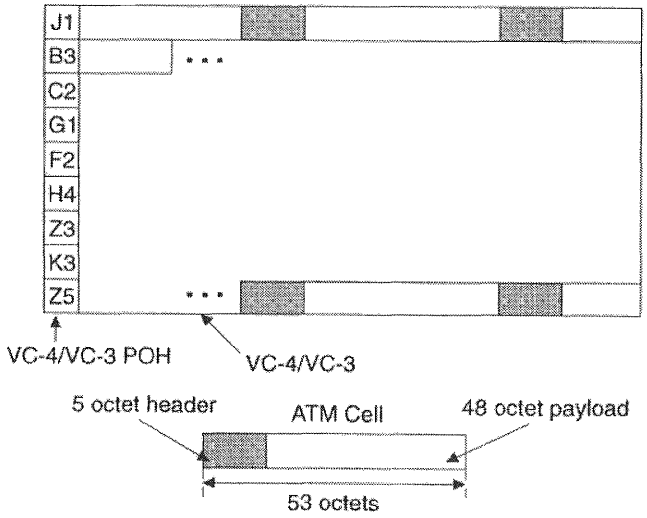
c) The SDH standard defines a series of containers, each of which corresponds to a PDH rate. These are basically time slots. PDH signals are mapped into their relevant containers with bit stuffing. Added to this is the Path Overhead (PoH) which allows end to end monitoring (BER) and management. The tabular format helps to synchronise the clock as each row contains 'fixed stuff' which is used to trigger the clock as each frame is processed. Hence the clock is sent along with the data payload and frames are still sent every 125usec even if there is no actual data to be sent.

1st basic SDH mux unit is the STM-1 (synchronous transport module) frame which is at a data rate of 155Mbits/sec. Higher mux is done by byte interleave by 4 (STM-4, STM-16). STM-1 signal is a repeated series of 125usec frames of 270x9 (2430) bytes. (Same as 1x64kbit/sec byte to avoid delays).

If you have 32 phone lines being digitised to 64kbits/sec then each channel will occupy one 8 bit bytes in ever STM-1 frame. Hence the payload will be 32 bytes per frame. The position will not be constant in every frame to help minimise latency between reception of frames and to aid the synchronisation process.

d) SDH has been designed for low latency and minimal delays, which is ideal for voice. Internet data tends to be bursty and is best suited to packet switched networks. This more difficult to map onto SDH, especially compressed video streams which are bursty and delay sensitive.

For this reason, the Integrated Services Digital Network (ISDN) standard was defined to provide digital access to the home. One possible solution is the section of B-ISDN called asynchronous transfer mode or ATM. Or the currently popular internet protocol (IP) which has global acceptance.



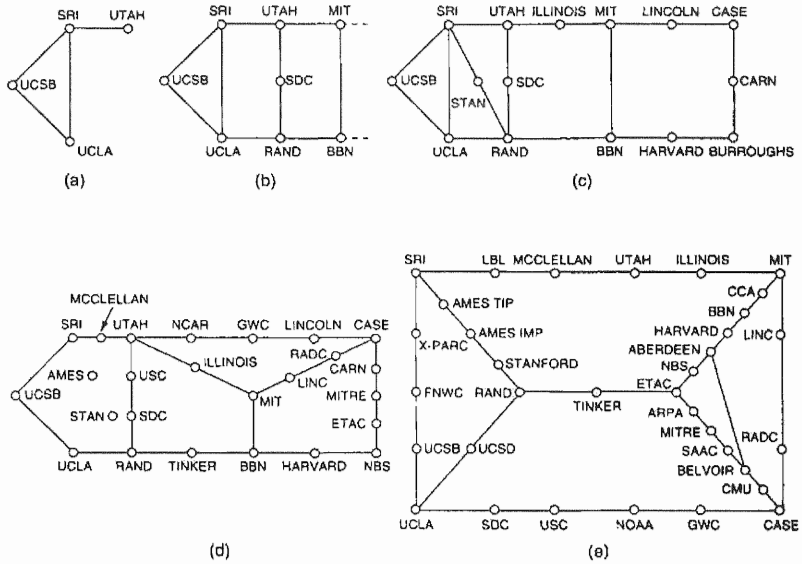
The advantage of the STM-1 frame is that it offers a huge wealth of virtual bandwidth as well as management services which will make ATM or IP over SDH easier than just pure ATM or IP systems. One of the problems with IP transmission over SDH is that the IP packets have a variable packet length between 40 and 10000 bytes, depending on the choice of physical technology. Mapping a variable length packet with SDH is difficult as each packet has to have its own POH assigned in every frame. There are 2 ways this can be deal with:

- Map a fixed packet size into the STM-1 frame which is large enough to take the biggest IP packet. This works well for Ethernet as the MTU is 1500bytes which can be easily set per frame. Any shorter frames will be padded out to 1500bytes which is inefficient use of bandwidth.
- Fragment each IP packet into ATM cells and then send as ATM data. (IP over ATM is fairly common in optical fibre networks).

Question 4

Crib (more verbose than expected from the students)

a) The internet emerged from a US government and military initiative to enable the interconnection of different, mainly UNIX-based computer systems for intercommunication. The origin of the internet can be traced to the mid-60s with the creation of the US military network ARPANET. This was designed to produce a network that was capable of operating after a nuclear strike, in other words, a network that was capable of distributing the routing of data and not rely on a single vulnerable controlling station that could be easily targeted. The structure of the network was also to incorporate the then radical concept of packet switching to transmit the data from node to node. There was also a strong desire to utilise existing transmission media such as network structures and telephones of existing operators. The idea was to piggy-back the network as much as possible.



(a) Dec 1969. (b) July 1970. (c) March 1971. (d) April 1972. (e) Sept 1972

The first network 'evolved' in 1969 between 4 US universities, mostly through the creative research of the staff in those four nodes. To make it more difficult, they each had different incompatible machines with which to set up the network. Yet somehow they did it and by late 1972, the network spanned the US. As the network expanded, they realised that the original ARPANET protocols were not suitable for 'internetworking' and so in 1974 they spawned TCP/IP.

A worldwide community of inter-linked computers quickly emerged. By 1990 the internet contained 3000 networks and over 200000 computers. In 1992 the millionth host was added and by 1995 there were multiple backbones, millions of hosts and tens of millions of users. The size approximately doubles every year. As the popularity of the internet has grown, so have the number of servers and routers making up the network. However, the reason for the rapid growth in the internet also provides a major challenge for the next stage of its development. The factor enabling rapid growth was the possibility to add further servers and routers to the network at almost any point in the network without considering a master plan. The traffic flows in the

network as a whole are therefore largely unmanaged and unmanageable. The only solution to slow response or congestion can be to add more capacity. Whether the capacity is added at the most appropriate point is a matter of chance.

b) Being a unique address, the IP address allowed an end user to be identified, no matter how many transit servers, routers or networks would have to be traversed along the way. Along with the IP address came the domain name which identifies each unique address.

There are various protocols that go to make up the transport control protocol/ internet protocol (TCP/IP) stack, which can be roughly mapped onto the OSI seven layer reference model. TCP/IP is one of the widest used protocols as it is the heart of the internet and allows world wide access to a common network.

<table>
<tr><td>OSI
7
6
5</td><td>X - windows</td><td>TELNET</td><td>FTP file transfer protocol</td><td>SMTP Simple mail transfer protocol</td><td>SNMP Simple network manage protocol</td><td>TFTP Trivial file transfer protocol</td><td>RCP Remote procedure call</td><td>NFS Network file server</td></tr>
<tr><td>4</td><td colspan="4">TCP</td><td colspan="4">UDP</td></tr>
<tr><td>3</td><td colspan="4">ARP ICMP IP RARP</td><td colspan="4">Gateway protocols BGP EGP</td></tr>
<tr><td>2</td><td colspan="4">SNAP LLC (eg Ethernet LAN)
SLIP (PPP) Serial Line</td><td colspan="4">Frame relay etc</td></tr>
<tr><td>1</td><td colspan="8">Physical network</td></tr>
</table>

It the heart of the stack is the internet protocol (IP) which is roughly equivalent to an OSI layer 3 protocol and the transport control protocol (TCP) an approximate equivalent of OSI layer 4. The mapping of TCP/IP onto the OSI model is not ideal, as there are some layer 2 processes done in both TCP and IP and IP has no real provision for end to end connection management.

The protocol indication determines whether TCP or UDP protocol is used in the next higher layer, and therefore determines to which protocol agent, the datagram should be delivered. (Very non-OSI). The source and destination addresses are both 32 bits. It is normal to write these addresses as four octets separated by dots (eg. 169.129.24.88). Each of the four decimal values may only have a value between 0 and 255.

The transport control protocol (TCP) is usually used in conjunction with IP and the internet control message protocol (ICMP) to guarantee reliable transmission. On an end-to-end basis, TCP ensures correct sequencing of arriving frames of data and requests retransmission when necessary. This is really a layer 2 function, but is not catered for well in IP. An alternative to TCP is the user datagram protocol (UDP), a simpler protocol which does not perform retransmissions. This is left to the application if necessary.

c) IP is fundamentally a connectionless protocol unless ICMP is used via the options fields. The problem is that ICMP is not considered on the fast path by most internet routers, hence the QoS attempted for a voice service cannot be met. In fact by removing ICMP packets off the fats path will add significant latency.

TCP can offer very complex services including acknowledged connection oriented service and even transmission via a form of virtual circuit to minimise delay and maximise quality of service (QoS),but this is on top of IP which is a connectionless protocol where packets are routed as they enter each network node. Hence QoS relies on other traffic not causing significant latency.

Protocols such as VoIP and SIP rely heavily on the fact that the bandwidth of a voice channel is very small and offers little overhead on a routers workload. As traffic increases, this becomes a problem and eventually the QoS is lost. One of the features of VoIP is the use of heavy compression codecs to reduce the bandwidth even further to avoid problems with congestion. Even with these techniques, the eventual overhead of handling many calls with limit router latency.

The solution most VoIP systems have taken is to install their own proprietary routing hardware which is VoIP enabled and optimised. This works well, but limits the choice of networks that can be used and is heading towards an internet based SDH mechanism and commercial basis.

Assessors' remarks for question 1: Part (a), which was almost universally well answered, asked candidates to describe branch hazards and resolution in the standard MIPS datapath. Part (b) asked candidates to suggest an improved design in which fewer instructions need to be flushed. Several candidates realised that the key was to move certain functional blocks earlier in the pipeline, and almost all realised that extra units at any pipe stage might require a slower clock. In part (c), candidates considered two dynamic branch prediction schemes. Most (but by no means all) calculated the correct asymptotic prediction accuracies. For the hardware implementation, it was pleasing that several candidates sketched out plausible logic designs, but very few discussed how to organise the state information for more than one branch instruction.

Assessors' remarks for question 2: This popular question examined aspects of I/O. Almost all candidates understood the difference between latency and bandwidth in (a), though some of the bandwidth estimates in (b) were wide of the mark. In (c), many candidates missed the point, and assumed that the word "cache" necessarily referred to the layer of storage between the CPU and main memory, whereas the question was of course referring to using main memory as a cache for the disk file. There was also a lot of discussion of DMA, which is of marginal relevance. Only the best handful of answers addressed the contention between the file system cache and conventional VM. (d) was generally well answered, though there was an unfortunate tendency to distinguish the two approaches in terms of parallel/serial and not bus/point-to-point network. Almost all candidates understood the significance of the 8/10 code in (e).

Assessors' remarks for question 3: This was a fairly standard question on the synchronous digital hierarchy. Most got the direct bookwork sections correct and showed they understood the evolution of plesiochronous to synchronous and the associated reasons. The majority missed the point of the frame format being 2D and the fact that it was related to the delivery of the clock within the frame every 125 μs.

Assessors' remarks for question 4: An internet question where the majority made a good effort to explain the evolution and scaling of the internet based mostly on book work. There were a variety of answers to the section on mapping TCP/IP onto the OSI model. The Assessor was impressed by the many different interpretations of layer mapping that were made. Overall a well answered question.

Andrew Gee & Tim Wilkinson
May 2009

Part IIA 2009

Module 3F5: Computer and Network Systems

Numerical Answers

1. (c) First scheme 80%, second scheme 90%.